(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
3 January 2002 (03.01.2002)

PCT

(10) International Publication Number
WO 02/01520 A1

(51) International Patent Classification⁷: **G07F 19/00**, H04L 9/32, G07F 7/10

(21) International Application Number: PCT/IB01/01120

(22) International Filing Date: 23 June 2001 (23.06.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

00810556.1	26 June 2000 (26.06.2000)	EP
01810482.2	15 May 2001 (15.05.2001)	EP
PCT/IB01/00896	21 May 2001 (21.05.2001)	IB

(71) Applicant *(for all designated States except US)*: **CO-VADIS S.A.** [CH/CH]; Chemin du Pont du Centenaire 109, CH-1228 Plan-les-Ouates/GE (CH).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: **HILLION, Hervé** [FR/FR]; Chez Bruchon, F-74570 Thorens-les-Glières (FR).

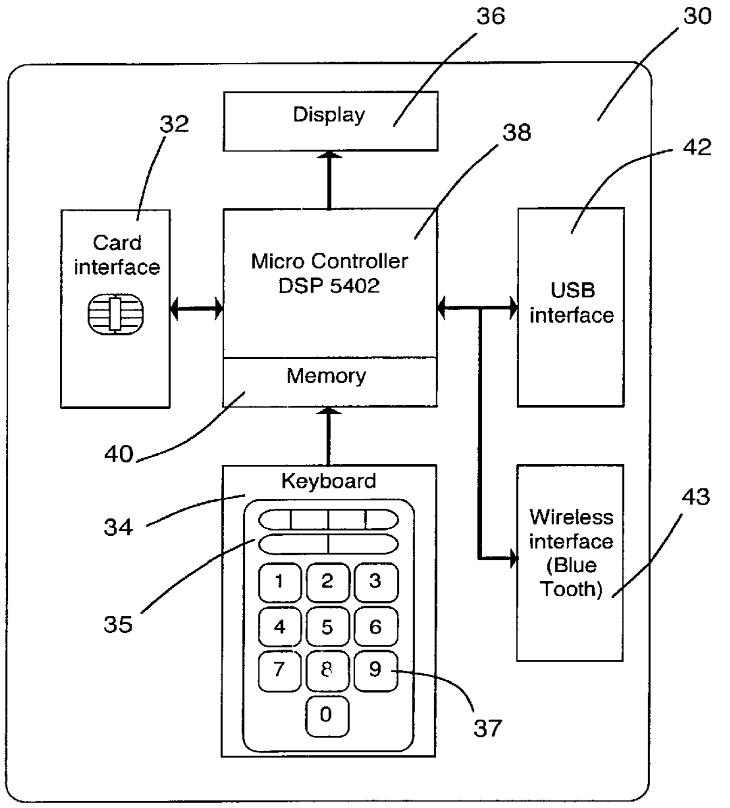
(74) Agent: CRONIN, Brian; Cronin Intellectual Property, Route de Clémenty 62, CH-1260 Nyon (CH).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,

[Continued on next page]

(54) Title: DEVICE FOR CARRYING OUT SECURE TRANSACTIONS IN A COMMUNICATIONS NETWORK



(57) Abstract: A device, in particular a card reader (30) connectable as a terminal of communications network such as the Internet for carrying out secure transactions, in particular using a protocol such as SET, comprises a keypad (34), a display (36) for displaying messages related to a secure transaction in progress, a DSP microprocessor (38) and a memory (40) comprising an E^{2}PROM and a RAM. The E^{2}PROM is arranged to store software in three shells, a boot shell (62), a system shell (64) and an application shell (66). The boot shell contains non-downloadable ground software that manages software download into the system shell and the application shell. The device is associated with downloadable system shell software and application shell software, one application being arranged to momentaneously store in a network-inaccessible part of the RAM a PAN or PIN code keyed in via the keypad (34); encrypt the code and output the encrypted code to the communications network. The device is useful for secure transactions in e-commerce, e-banking, e-tax, e-mail, telecommunications and restricted access systems, etc.

WO 02/01520 A1

IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*

Device for Carrying out Secure Transactions in a Communications Network

Field of the Invention

This invention relates to a device connectable as a terminal of a communications network for carrying out secure transactions or any PIN code processes in the network, for user ID operations with or without a smartcard, in particular in conjunction with operations like e-commerce, e-banking, e-tax and e-mail using the Internet. The invention is applicable *inter alia* to secure transactions in a communications network like the Internet wherein encrypted data is communicated between a card issuer system, merchant site applications and a banking system using a protocol such as the Secure Electronic Transaction (SET). The invention also relates to a cardholder system for carrying out secure transactions in a communications network, comprising a device associated with a local computer application (LCA), such as a PC, to which the device is connected.

Background Art

There have been many proposals for carrying out secure transactions in an insecure communications network like the Internet.

For example, channel security schemes such as secure HTTP (S-HTTP) and the Secure Socket Layer (SSL) protocol are intended to create confidence between two communicating parties. S-HTTP uses digitally signed messages with a heavy encryption key to ensure security and authentication. SSL utilizes digitally signed certificates to provide authentication and security by heavily encrypting the messages. This channel security technology makes confidential transmitted information, but it does not protect the merchant and the bank involved in a transaction against "true-false" card numbers, nor does it protect the customer against the abuse of their personal data, and it does not allow the bank to guarantee payments to merchants.

Multi-party protocols have also been proposed for credit transactions, like Secure Transport Technology (STT), Internet Keyed Payments (iKP) and Secure Electronic

Payment Protocol (SEPP). The aforementioned SET secure payment technology represents the state-of-the art in Internet based payment processing. More recently a new specification known as EMV, based on the SET protocol, has been developed by EuroCard-Mastercard-Visa with plans for future implementation.

The SET protocol is designed on a hierarchy of trust for the management and verification of SET certificates by certificate authorities, notably between a Cardholder Certificate Authority (who issue cards to cardholders), a Merchant Certificate Authority (who issue certificates to merchants) and a Payment Gateway Certificate Authority, controlling a SET Payment Gateway. The SET protocol is based on the exchange of digital certificates to provide a banking guarantee. SET prevents fraudulent use of payment card numbers, and prevents the interception of confidential information by a third party.

These protocols provide good security between the banks and merchants but, like the previously-mentioned channel security schemes, do not provide full security to the individual user. Although the user's private data like a PAN or a PIN code is transmitted through the Internet in encrypted form in such a manner that the encrypted data can only be decrypted by an authorized institution, when the user enters this information for example via a PC connected in the network, the unencoded data is exposed to an "attack" by an unauthorized person. This has led to abuse and to a considerable amount of litigation concerning unauthorized transactions.

Research over the last 4 years shows that electronic data exchange and payments on the Internet are still being held back by concerns over lack of security and privacy. Recently, the use of the Smart Card combined with a safe protocol (Secure Electronic Transaction SET), called Chip Card-SET (C-SET), designed to provide users and merchants with digital certificates has been recognised to be the obvious option for building the base of any secure transaction.

WO 98/4965 describes an Internet payment system using a stored-value card, which generally describes the encryption protocols for transactions with a merchant server and a payment server.

Card readers for accepting so-called smart cards (also known as chip card, IC card or ICC, hereinafter referred to collectively as "ICC" or simply "card") are also known. These card readers typically comprise a keypad, a display for displaying messages related to a secure transaction in progress, a microprocessor and a memory.

WO 98/07255 describes a transportable authentication communication device with a compact housing possibly serving as a card reader, having a key-pad for entering a PIN and a display. The transportable device incorporates a cryptographic module containing all of the necessary circultry for encryption and authentication as well as a modem.

However, the known card readers do not overcome the above-indicated problem and generally each type of card reader has the limitation that it is specific for a particular application.

EP 0587375 describes a different type of device, which is a security unit connected between a PC and its keyboard, the PC storing several programs which allow it to be operated in a transparent mode or a special handling mode, i.e. with encryption/decryption or computation of a message authentication code. This device aims to avoid duplication of keyboards/displays. It is designed to act as a key and is not adapted for carrying out transactions like purchases using merchant site applications using secure protocols like SET.

US patents 6,920,730 and 6,056,193 relate to computer keyboard units incorporating a card-reader interface, having control circuitry whereby key codes generated by actuating keys are either passed directly to a host computer, or to the card-reader interface so a PIN can be directed to an inserted smartcard without the PIN passing out of the keyboard unit.

In US patent 5,920,730, the computer keyboard changes from normal mode to secure mode bypassing the host computer to input PIN code directly into a smartcard received at its ICC interface. This device is not designed for, and is not capable of, carrying out secure transactions on the internet. Instead, its purpose is to keep the PIN in the keyboard. If connected to the internet, an input the PIN could be accessed through the PC's basic input-output system (Bios).

US patent 6,056,193's computer keyboard with integral card reader also aims is to retain the PIN in the keyboard for security purposes. This keyboard is also not designed for and is not capable of carrying out secure transactions on the internet. Its purpose is the opposite, to keep the PIN within the keyboard. No PIN encryption is contemplated, only the transmission of encryption keys and other types of personal data. Again, if the device were connected to the internet, security would be corrupted by accessing the PIN through the PC's Bios.

UK-A-2274523 describes a portable electronic fund transfer device with keyboard and display, a magnetic swipe card reader and a DTMF circuit.

US 5,336,870 describes a payment terminal with a magnetic card reader, a display, a modem and encryption means, connected to a host computer for managing payments.

Summary of the Invention

An object of the invention is to provide a device connectable as a terminal to a communications network such as the Internet for carrying out secure transactions or any PIN code processes over the network providing full security for the user in connection with user ID operations with or without a smartcard, and which can be used with different applications and for many different secure transactions, for example in conjunction with operations like e-commerce, e-banking, e-tax, e-mail, telecommunications and restricted access systems, making it very versatile.

The invention provides a device connectable as a terminal of a communications network for carrying out secure transactions or any PIN code processes in the communications network, the device comprising a keypad, a display for displaying messages related to a secure transaction in progress, a microprocessor, advantageously a DSP microprocessor, and a memory, comprising a non-volatile memory such as an E^{2}PROM and a RAM.

The device's non-volatile memory (hereinafter exemplified as an E^{2}PROM) is arranged to store software in three shells, a boot shell, a system shell and an application shell, wherein the boot shell contains non-downloadable ground software that manages boot operation and software download into the system shell and the application shell.

The device is associated with software downloadable into or downloaded in the a non-volatile memory, namely system shell software containing operating system software that manages the application shell, an ASN library and an encryption/decryption toolbox and optionally a card manager; and application shell software containing software that manages applications for secure transactions.

The downloadable application shell software comprises an application that, when the software is downloaded into the non-volatile memory and the device is connected to the communications network, is arranged to perform a code encryption operation internally of the device when a user keys in a code via the keypad to perform a secure transaction. This code may for example be a PAN or PIN code whose entry constitutes a necessary step for transaction acceptance. This application is so arranged that, when the software is downloaded into the non-volatile memory and the device is connected to the communications network, it encrypts within the device a user identification code keyed in via the keypad and outputs the encrypted code, without the keyed-in code being accessible from outside the device.

For example, when the user identification code is keyed-in, the unencrypted code is firstly momentaneously stored in a non-readable part of the RAM, inaccessible from the network. The momentaneously-stored keyed-in code is encrypted (the entire encryption usually being carried out by means of the downloaded encryption toolbox) then erased from the RAM soon after encryption. The encrypted code is then output from within the device to the communications network, for example in response to the user entering an acceptance code. This enables acceptance of a secure transaction by the user keying in a PAN and/or a PAN or PIN code, without the unencrypted PAN and/or PIN code ever being accessible from the communications network.

Moreover, the device is simple to use and can be produced at low cost notably by using the DSP technology, making it possible to use the device for macro and micro payments, and a wide range of secure transactions.

The device according to the invention is a veritable lock allowing a security check on a secret code (for example a PAN code associated with a magnetic card or a smart card, or a PIN code associated with a smart card), without any transmission of the unencrypted secret code to the communication network, e.g. via the CPU of a personal computer connected to the Internet and to which the device is connected.

At least part of the cryptography takes place within the device: the encryption signature is generated within the device, and the PAN and/or PIN is/are encrypted within the device. When the transaction on the network is protected by a protocol such as SET, the transaction can be signed also by the device. The SET issuer is able to authenticate the device, and the issuer has insurance on the security level. It is possible but however not necessary to manage the SET certificates inside the device.

A large amount of memory can be made available in the device (say 8 Mbits), so many applications can be installed.

One of the main advantages of the device is the speed of the transaction due to the DSP processor used. This performance allows a high level of protection of data transferred from or to the network (typically via a connected PC) by undertaking huge calculations for data encryption or decryption. For example, the device is able to encrypt data by using an RSA public key of 1024 bits with its full exponent in less than 5 s.

Moreover, the DSP processor is able to provide virtual emulation, in real time, of functions such as that of a modem or more generally of networks interface functions, making use of the downloaded software. Consequently, the device according to the invention dispenses with the need to incorporate a modem or other interface hardware to perform such functions.

The device is advantageously used in connection with an ICC to perform secure transactions on a communications network, the device enabling on-line card authentication. For this purpose, the device preferably comprises a card reader for an ICC card, the device being arranged to enable card identification and acceptance on an ICC inserted in the reader followed, if the card is accepted, by the above-described operations to establish transaction acceptance. At least a part of these card identification and acceptance operations and/or of said transaction acceptance operations can advantageously be performed using a processor incorporated in an ICC inserted in the device, as will be more fully described below.

In this context, the merchant is certified whereas the device according to the invention operates in a closed security environment in which the device provides certitude of authentication of the purchaser by electronic signature. On-line authorization of the transaction provides a guarantee for the payment or other transaction. Control of the card signature takes place on line, not off line. Bogus merchants/bogus cardholders are eliminated. There is no possibility of fraud, therefore no repudiation, no litigation with the cardholder and no chargebacks to be managed.

An important feature of the device is its ability to download system and application software and to use the downloaded software for performing the operations involved in establishing a secure transaction with the communications network. Particularly in the banking sector, there are frequent modifications liable to make software become obsolete rapidly. The device's downloading capability enables constant updating according to software developments, in particular new or upgraded applications can be downloaded into the device from a server or from a local computer application connectable to the communications network.

For the purpose of allowing software upgrade, the device's non-volatile memory (E^2PROM) can comprise a buffer zone serving as a buffer memory for temporarily storing previously-loaded system and/or application software during a download operation, the memory being associated with means, responsive to a signal indicating interruption or failure of a download operation, for reloading system and/or application software stored in the buffer memory.

A device according to the invention can be incorporated in various pieces of equipment, one particular embodiment being an external card reader connectable to a personal computer preferably via an USB connector. Such card reader can include an ergonomically-designed card reader body that can be held in the hand or placed on a support surface during use, the body having a slot for receiving an ICC leaving a part of the ICC protruding during use to prompt card removal when an operation is finished. The card reader body also integrates the keypad and display, and internally contains the memory.

Another embodiment is a discrete keyboard of a personal computer incorporating a device according to the invention, the keyboard being connectable to a personal computer via an USB connector. The main set of keys of the keyboard may be used also as the device keypad, or the device can have a dedicated keypad (such as a standard numeric keypad) located alongside the keyboard's usual keys. The keyboard may advantageously incorporate its own secure display for the messages related to secure transactions, or the computer screen can be used for display. It will preferably also have a slot for receiving an ICC, the keyboard thus also serving as a card reader.

This embodiment of the device according to the invention, which is further described and claimed in copending patent application PCT/IB01/00896, is to be contrasted with the security unit of EP-A-0587375 which is connectable between the PC's CPU and its keyboard for selectively using or not using an encryption mode.

Use can thus be made of the computer's keyboard or just its numeric keypad for entering the PIN. Such keyboard has an ICC interface and is associated with a keyboard matrix selectively connectable between a normal operation mode for communicating keycodes to the Bios (Basic Input-Output System) of a locally connected computer, and a secure transaction mode for carrying out secure transactions or any PIN code processes with or without an

ICC inserted in the ICC interface. In this way, the keyboard can be used in two modes, a normal mode where generated key codes pass to the computer interace in the normal way (without encryption) and a PIN mode where the generated key codes are subjected to a PIN entry protocol and encryption, whereas the device's microprocessor and ICC interface are inaccessible through the Bios of a connected computer.

The device according to the invention can also be incorporated in a portable personal computer, in which case the device must have its own memory and keypad separate from the computer's hardware accessible to a communications network. The PC's screen can however also be used as the device's display.

Further examples are a Set Top Box of a digital television receiver connectable to a communications network by a parabolic antenna, a mobile communications device such as a telephone connectable via a cellular system of via satellite to a communications network, a fixed telephone set, and a point-of-sale vending device such as a distributor for rental video cassettes, all of which can incorporate a device according to the invention.

The invention also concerns a cardholder system for carrying out secure transactions in a communications network, comprising a device as set out above associated with a local computer, such as a PC, to which the device is connected, preferably by a USB connector. In such a cardholder system the local computer can store software including a Dynamic Link Library (DLL) for cooperating with system shell software and application shell software downloaded in the device's non-volatile memory to perform secure transactions in the network.

Another aspect of the invention is a communications network in which encrypted data is exchanged between a card issuer system, merchant site applications, a payment gateway and a device or a cardholder system as set out above, wherein the device is connected to the communications network to form a terminal for carrying out secure transactions in the network, in particular transactions in which the card issuer system, merchant site applications, and payment gateway exchange electronic certificates according to a protocol such as SET. The device can provide particularly high security by combining security based on the ICC with the certificate principle embodied by SET and other protocols. Using the device, there is no circulation of banking secrets or personal secrets on the Internet, only enciphered data.

The device's downloading capability makes it particularly versatile in the context of security protocols like SET, because the device can be constantly upgraded relative to a particular protocol, and can be upgraded by simple software download if a new specification like EMV or the "Blue Tooth" standard comes into use.

The device of the invention is advantageously used with an ICC insertable in the device for effecting a secure transaction in a communications network, but it can also be used for effecting secure transactions in a communications network without the insertion of an ICC in the device. Examples of transactions without using an ICC are e-mail encryption, and operations where credit card data such as the credit card number and expiry data are keyed in by the user, possibly associated with providing other means of identification, as required. Even when the device is provided with a card reader, it is possible to use the device for secure transactions without inserting a card therein.

The device can be used in general for all secure transactions with a communications network, including credit card and debit card applications, virtual card transactions, as an e-purse (in particular for reloading e-purses at home), for health cards, e-trade, e-banking including home banking, e-mail security, secure access to personal data, secure dealings with public authorities including e-tax, e-voting including voting as company shareholder, telecommunications systems with fixed telephone sets and mobile telephones, restricted access systems including defense-related and private access systems, etc.

The invention also concerns a computer program, stored on a computer readable medium such as a CD-ROM or on a server, for carrying out secure transactions in a communications network to which a device according to the invention is connected via a local computer application. This program includes the above-described system shell software and application shell software downloadable into the E^2PROM of the device, as well as software downloadable into the local computer application for managing all transmissions between a local computer and the device, including a Dynamic Link Library (DLL), a system driver for USB management and for I/O management, an executable file for managing the download operations, and optionally a test program.

Another aspect of the invention is a device, in particular associated with an LCA, configured to produce an encrypted digital signature derived from a private key associated with the device, a user identification code, a private key associated with the device's manufacturer, and a public key of the device, and in particular is arranged to encrypt the public key of the device before transmission to produce a transmittable digital signature containing no public key. The device and an associated LCA can also be configured to receive and authenticate such digital signature.

Brief Description of Drawings

In the accompanying schematic drawings, given by way of example:

Fig. 1 is a diagram illustrating a device according to the invention in the form of an ICC card reader;

Fig. 2 is an overview of a system for secure transactions over the Internet;

Fig. 3 illustrates the software architecture;

Fig. 4 is a diagram illustrating a sequence of operations involved in a secure transaction;

Fig. 5 is a similar diagram illustrating a sequence of operations involved in a download operation;

Fig. 6 is a block diagram showing the functions performed by a sender using a device (in the given example a keyboard unit) to perfom a secure transaction with a digital signature; and

Fig. 7 is a block diagram showing the functions performed by a receiver to authenticate the digital signature.

Detailed Description

The device shown schematically in Fig. 1 is a card reader 30 that has been designed to perform all card operations and the application processes that must be protected from the external world of the card reader.

The card reader 30 comprises a card interface 32, a keypad 34, a display 36 for displaying messages related to a secure transaction in progress, a DSP microprocessor 38 and a memory 40 comprising an E^{2}PROM and a RAM. The keypad 34 and display 36 constitute a user interface. The device further includes an USB interface 42 for connection of the reader to a personal computer 44 via an USB connector 46, see Fig. 2. The reader hardware connects to the PC 44 as a high-speed bus-powered device on the USB bus.

Optionally, or alternatively, the device includes a wireless interface 43 for connection to a network using for instance Bluetooth wireless technology, details of which are available at http://www.bluetooth.com.

The keypad 34 includes a set of function keys 35 and ten digit keys 37 for keying in the digits 0 to 9. The reader's function keys 35 can for example include: a language key for switching the language of the displayed message; a conversion key for displaying currency conversions (e.g. FF and Euro); a cancellation key for interrupting the current transaction; a correction key for erasing the last pressed key (correction of the introduced PAN and/or PIN code); and a validation key for accepting the entered PAN and/or PIN code and to continue the payment process. Further functions can be included, as required.

System Overview

As illustrated in Fig. 2 the reader 30 with an inserted ICC 31 is connected to a computer 44 that uses an Internet browser application. The computer 44 user can open a session on a merchant site 50 that offers a service or product, the purchase of which will be controlled using an exchange of electronic certificates with a merchant bank 52 and an issuer bank 54, using for example the SET protocol, as described in greater detail later.

By activating a service at the merchant site 50, the user starts a merchant site application MSA, also designated by reference 50. This operation can connect other Internet sites and start corresponding applications on many sites. One of these applications will send back to the user computer 44 a so-called "Wake-Up" request that will activate a specific application on the local computer 44. This application activates the reader 30 and manages the transaction between reader 30 and the remote Internet site. When the operation is finished, the local application cancels the link to the reader 30 and stops. The remote site receives a positive or negative acknowledgment after the operation.

The reader 30 and the local computer 44 represent the cardholder system, as defined in the SET standard. The ICC 31 and the issuer bank 54 are the issuer system. The merchant site 50, its bank 52 and the associated payment gateway represent the acquirer system.

The reader 30 has been designed to be used by any computer application, typically a personal computer 44. The reader 30 is activated using a dynamic-link library (DLL) stored in the computer 44's hard disk, and calling the specified interface functions. As soon as these operations are done in a correct way, the reader 30 performs the requested applications. Each application is independent and its software can be downloaded separately into memory 40's E^2PROM. This modular implementation allows the reader software to be updated or even upgraded.

The reader 30 is connected to the computer 44 through the USB cable 46. The reader 30 is identified by the local computer 44. If not, the operating system of the computer automatically starts an installation procedure, after which the reader 30 is identified by the host.

As described previously, the user by activating a merchant site 50 will download a Wake-Up request. This input file activates a local application, called the Local Computer Application LCA that will manage the reader 30 on the local computer 44 (the reference 44 is hereinafter used also for the LCA). This LCA 44 is linked to the reader 30 by a Dynamic-Link Library (DLL) that will transmit the LCA requests to the reader. The LCA 44 also formalizes the reader responses and sends them to the payment gateway through the MSA 50.

When the transaction is terminated the payment gateway (of Merchant Bank 52) sends to the MSA 50 and to the LCA 44 a positive or negative acknowledge message. When it receives this message, the LCA 44 stops the reader 30 and exits. In case this message does not come, a timeout is activated in the LCA 44 which stops the reader 30 and the local application after having displayed an error message.

The merchant site 50 corresponds to a web address that is usually described by a URL address, whereat a set of products can be bought. The products to be purchased can be selected by adding them to a basket. At the end of this selection, the purchaser can prepare a purchase command and select the payment support (SSL credit card operation, card operation with a virtual wallet, etc.).

The reader 30 is protected from being attacked from external entities by the following features.

A specific application can be provided for transparent commands from the local computer 44 to the ICC 31 based on the PC/SC specifications. However, no transparent command for the ICC 31 is allowed between the computer 44 and the reader 30 once a secure transaction has been initiated. For example, it is not possible to enter the PAN or PIN code on the computer keyboard and transmit this data to the reader 30 for checking. Confidential information like the PAN and/or PIN code is always encrypted before being sent to the computer 44 and vice-versa. The sensitive code lines are scrambled before being written in the reader 30's E^2PROM.

Security is based on cryptogram algorithms: the higher the calculation performance, the higher the security level. However, a practical restriction on the cryptogram calculations comes from the real time operations. A calculation duration over a dozen seconds is unacceptable for the user. With the described reader, encryption calculations can be performed in under 5 seconds.

The SET protocol is advantageously chosen for the data management in order to match security requirements. Management of the SET protocol requires hardware and software with the following capabilities. A large memory space for the I/O buffers, larger than 4 kBytes in standard use, and extendible to say 64 kBytes for special cases. A flexible management of the I/O buffers is necessary, because all data could have various lengths. The I/O structure of the data should be managed in the heap of the reader's RAM, which should be larger than 4 kBytes. There must be a capacity for a large amount of calculations, which requires a DSP 38 with sufficiently high processing speed. This requires a RAM of at least 16 kWords. For example, a DSP of type TMS320UVC5402 available from Texas Instruments can be used.

Message Authentication Code can use up to 2048 bits of RSA coding (Rivest-Shamir-Adleman algorithm) with generation of private keys. A hash coding process of the signature can be provided.

The reader 30 is relatively low-cost, by using a low-cost DSP and E^{2}PROM, and is robust and easy to install. It is not possible for the user to erase essential data by performing a wrong manipulation.

Moreover, as mentioned previously there is no need for the reader 30 to include a modem or other interface hardware, because the interface operations can all be carried out in real time by virtual emulation, using the DSP 38 and downloaded software.

Taking into account that the invention is directed *inter alia* to the banking market segment having multiple functions that are frequently modified, the software is likely to become rapidly obsolete. The reader 30 is accordingly designed with a download capability. Only basic essential software remains in the reader 30. The drivers (for card 31, keypad 34, etc.), the libraries (ASN.1 messages, etc.) the system shell and all the applications are downloadable. This requires the following capabilities.

A large reserve of E^2PROM memory is provided for allowing the old version of the downloaded software to be buffered. In case of an interruption of the download operation, a restart will reinstall the old version. The download function should be triggered by an external application through the LCA 44 or by a user request. Moreover, the software must be controlled before being written in the memory.

For the protection of data, some essential information must be protected from being read by external people. For this, a part of the RAM is in a non-readable zone.

The reader 30 is associated with a DLL to provide an API to the user application, and a USB driver for the DLL to communicate with the system's driver stack. It also has an executable that manages the firmware download to the reader, and a test program that performs factory tests. This PC software is supplied with the reader for example for Windows 9872000 platforms. At the end of testing the reader receives a signature that cannot be removed.

The DLL is copied from an installation CD to the PC hard disc during the card reader installation when the system recognizes a new USB device, and is loaded as required by the user application. The API to the user application consists of three functions to open, read/write and close the reader 30. There are additional (non API) functions implemented in the DLL to assist the card reader in its operation. A block-oriented asynchronous half duplex transmission protocol has been defined to transfer data between the DLL and reader 30.

The USB driver (supplied with reader 30) is copied from an installation CD to the computer hard disc during card reader installation when the system recognizes a new USB device, and is loaded as required by the system when reader 30 is connected to the USB bus. The DLL communicates with reader 30 through this USB driver.

The use of an E^2PROM in conjunction with a USB connection has the advantage that the reader 30 can operate from the mains supply without need for a battery.

The download executable (supplied with reader 30) is copied from the installation CD to the PC hard disc during card reader installation when the system recognizes a new USB device. It is executed from the LCA (via the DLL) when the user or a remote site application sends a download request. This download concerns updated firmware that should be transferred from an Internet site into the card reader. During this operation a window appears on the PC screen showing the transfer progress and status.

A factory test program is supplied for testing the card readers 30 after assembly. This program does basic functionality checks of the reader (USB, display, keypad, card insertion) and writes into its flash memory (E^2PROM) a coded serial number from a barcode label on the reader 30.

Software

The software installed in the reader 30 is essentially written in an object oriented programming language such as C and C++. But the reader processor 38, which is a DSP, only interprets assembler commands. For this reason the source files are firstly interpreted in assembler using a C/C++ compiler. Some functions that are critical in slowing the performance are optimized in assembler, for example functions that are called during the RSA calculations.

The software architecture is shown in Fig. 3. The reader software 60 has been built in three main shells, boot shell 62, system shell 64 and application shell 66. The boot shell 62 corresponds to the ground software that manages the software download; this shell can never be downloaded. The system shell 64, which is downloadable, corresponds to the rest of the operating system that manages the application shell. The application shell 66, also downloadable, contains the code that manages applications like the payment application.

As schematically illustrated, the boot shell 62 contains the USB manager, and all software for managing downloading, including an RSA/DES manager used for download. The downloadable system shell 64 contains an ASN.1 library which serves as I/O interpreter, the RSA and

DES encryption/decryption tools and the drivers, like the card, keyboard and display manager. The downloadable application shell 62 contains administrative applications like a terminal identification application and a download application. It also contains ICC applications, i.e. a set of applications for each chip card, for example controlling different types of payment, for example payments using EMV parameters, or prepaid card payments or so-called Mondeo payments.

The DSP 38 is powered when the USB cable 46 is connected to computer 44. It starts to run and enters the boot shell 62 where all the basic start-up procedure is performed. This procedure essentially checks the validity of the system and activates the USB bus connection. Then the reader 30 waits for an open request from the LCA 44.

Once the LCA 44 has activated reader 30, the DLL sends a message to reader 30. Depending on the type of the request, the message will be managed in the boot shell 62 or transferred to the system shell 64 and managed by the system shell, or transferred to the application shell 66.

The boot shell software 62 can never be upgraded. In one embodiment, the system and the application can only be upgraded together. In another embodiment, the system and the application can be upgraded separately.

The boot shell 62, system shell 64 and the application shell 66 each has its own version number. A hardware version number is written on the electronic plate of the reader 30. Because this number is known during the manufacturing process, it is reported in the boot software that is installed at the end of the manufacturing process. So a new hardware version (n_H+1) corresponds to a new boot version (n_B+1). If the modifications in the boot shell 62 are sufficiently minor that the new boot version is compatible to the previous system version, the new boot version will keep the same model number. So a new boot version does not always result in a new model number. The hardware, the boot and the system version are respectively numbers (n_H+1), (n_B+1) and (n_S) at the end of the upgrade.

As a second example, the actual hardware is kept but the installed boot shell is modified. This modification alters the system, that is able to take into account the two different boot versions, by implementation of a switch that executes a specific function depending on the boot version. In this case we have (n_H), (n_B+1) and (n_S+1) for the hardware, the boot and the system versions respectively at the end of this upgrade. Because the new system version is still compatible with the old boot shells, the new system will be integrated as a new software version by the download server. The previous model number is kept.

If the boot shell modifications are sufficiently important so as to be no longer compatible with the current version of the system shell 64, the new system version will correspond to a new model number. The new system version will not be downloaded as a new version to the previous system shell. In this case, the new model number is maintained separately. The system and application shells 64,66 are no longer compatible and must be selected by the download server. It is understood that creating a new model has the drawback that a separate and independent release support is needed for each model. This is fully transparent for the user but has drawbacks for the producer.

When a hardware modification requires a modification of the boot shell 62 that does not alter the compatibility with the current or previous system and application shells 64,66 (for example the reader's box colour is modified), it is not considered as a new model, but as a new sub-model that does not require a specific maintenance from the download server.

An upgrade of the boot 62 or the application shell 64 often leads to an upgrade of the system shell 66. But some upgrades do not require a system upgrade. So the system shell is still independent from the other shells. If modifications to the other shells alter the system shell 66, a new system version will be given that corresponds to a new application identifier sub-version number issued by a controlling Organization.

Like the system shell 66, the upgrade of the applications is still independent from the other shells and the other applications. The applications are always independent from one another. An application upgrade always gives a new application version number. If the application modifications correspond to a specification upgrade, a new application identifier main version number is given. If they correspond to an internal (manufacturer) upgrade, only the identifier sub-version number sub-version number of the issuing Organization is incremented.

The code contained in the boot shell 62 essentially initializes the DSP 38 and checks if a valid system is present. If yes, it starts the display, keypad and card processes. If not, the reader 30 waits about 5 seconds, then restores the previous system by re-installing its back-up copy. Then this procedure activates the USB connection 46, and indicates "Reader ready" when the USB checking is finished. At this time, the reader program enters into an infinite loop that corresponds to a state machine, wherein the following events can produce a change of state: an incoming message from the DLL (via USB 42); pressing of a key by the user (keypad 34); insertion of a card 31 (Card I/O 32). All these events are managed by interruption vectors.

In the boot shell 62, the card I/O is used only for the detection of the card insertion or removal. Code enters the system shell 64 when the following functions are called by the boot shell 62:

StartMainSystem	Starts and initializes the system shell 64.
DisplaySystemManager	Manages the display of the messages in the boot shell 62.
MainSystem	Transfers an incoming message to the system shell 64.
StopMainSystem	Stops the system.

The first and last functions control the execution of the system shell 64. The second function provides to the boot 62 the message library that can be upgraded. The *MainSystem* function controls all the access to the system, which operates according to the input message type. Messages that can enter the system shell can be split in four categories, the ASN (Abstract Syntax Notation) frame; Test requests; Download application requests; and Complementary Requests.

The input message, encoded according to the ASN specifications, is decoded and, depending on its command header, is treated by the application shell 66. The available applications are described below.

There are three entry point functions for each application that allow to initialize, to perform and to close the application. All of the data that are transferred inside the ASN structure of the I/O messages are managed by the system shell 64. These data are allocated and deallocated in the system shell 64, but managed in the application shell 66. The data that are specific to the application are allocated and deallocated inside the application shell 66.

Test requests concern manufacturing tests to ascertain that the reader 30 performs properly. Download application requests are managed by the boot shell 62, but the request for triggering the download application and its error management must be downloadable. For this reason, this part of the application is implemented in the system and in the application shells 64,66. Some specific application requests are managed by the system shell 64, notably to manage problems associated with management of huge I/O frames. Applications are called by the system shell 64. By definition, the applications should be independent from one another; they can depend from the boot and system shells; and they are called by three generic functions that are used for:

Initialization : Allocation of the data structure that is required during the process, initialization of the data and set up of the variable that controls the application process.

Processing : Non-blocking or blocking processing of the application steps.

Termination : Deallocation of the application data structure and final processing, like clearing of the display or request for card removal.

For example, the implementation of two applications is envisaged, an administrative application that manages the reader identification and the software download request; and a payment application that performs payment operations for a specific type of smart card.

As soon as a download operation is started, the previous system or/and application zone is copied in a backup or buffer zone of the E^{2}PROM and is erased. During this operation, the keypad 34 and the card I/O 32 are deactivated and reader 30 displays at 36 the message "Downloading. Please Wait". Before downloading, the reader 30 prepares its ASN.1 answers. This is done in two functions called in the system shell 64. A first function prepares the correct answer that is expected by the LCA 44, which contains the path of the DLL file when this library must be downloaded by the LCA 44 at the end of the reader download process. A second function contains error messages that must be returned in case of an error.

Secure Transaction Processing

Fig. 4 illustrates a secure transaction such as a payment carried out with an ICC 31 in an interface device (IFD), namely reader 30 using a local computer 44 as an LCA Wallet communicating via the Internet with a merchant at a merchant site MCA 50 and a Payment Gateway 55, using for example the SET protocol. In Fig. 4, the numbers 1 to 21 are used to designate successive steps of the transaction, as follows.

Steps 1-3: Reader Identification

The user by activating a merchant site 50 will download a *Wakeup* request as step 1, providing an input file that causes the LCA 44 to send a request for reader identification *RIDReq* to the reader 30 in step 2. Each reader 30 is identified by its coded serial number contained in its boot shell 62. If a recognized reader 30 is connected, an acknowledgment/response *RIDRsp* is returned to the LCA 44 in step 3.

Steps 4-7: ICC Identification

In step 4, the LCA 44 sends a Card identification request *CIDReq* to the reader 30 which transmits this request in step 5 to the ICC 31. Identification of an ICC 31 inserted in the reader 30 takes place using the ICC's internal processor, i.e. without using the encryption/ decryption toolbox downloaded in the reader 30. Once the ICC 31 is identified an acceptance message *CIDRsp* is returned in steps 6 and 7 to the LCA 44. In this ICC identification process, the reader 30 acts as a "window": all necessary encryption/decryption operations take place in the ICC 31 and in the LCA 44.

The *CIDReq* asks for the ICC identification and also retrieves the initial payment parameters. This command is used by the LCA 44 to obtain the payment parameters required for building the payment request initiation message.

Steps 8-17: Payment Transaction

A payment initiation request *PinitReq* is then transmitted in step 8 from the LCA 44 to the Merchant Site 50 and returned, accompanied by the necessary certificate, to LCA 44 in step 9. This payment initiation request *PinitReq* is normally followed by a payment request *PayReq*, so no message requesting card removal is displayed when it is completed. An error during the treatment of this request will interrupt the payment procedure, and display at 36 a message inviting card removal.

In step 10, the LCA 44 sends a payment request *PayReq* to reader 30. The *PayReq* asks the reader 30 to perform the payment process. When the reader 30 receives such a request, it cannot perform another operation until the payment application has been completed. The *PayReq* asks the reader 30 to perform a transaction that is secured with a card signature. Through this command, the computer 44 receives data and information that can be displayed and returns the encrypted sensitive data to the gateway 55.

Payment operations are carried out between the reader 30 and card 31 in steps 11 and 12. At this stage, the reader 30 checks if a card 31 has been inserted. If not, it requires card insertion. Then it checks if the inserted card corresponds to the one that has previously been identified during the *CIDReq* operation. If not, an error message is displayed saying that the card is invalid and the payment procedure is interrupted.

After this initialization step, the reader 30 displays the amount to be debited and asks the user to enter his/her PIN code. By entering and confirming a valid PIN code, the user accepts the payment transaction conditions. The reader 30 confirms the correct PIN code introduction and performs the protection calculations. Then reader 30 sends a payment response *PRsp* to the LCA 44 in step 13. At the end of this command, the LCA 44 receives all the sensitive data encrypted in a protected frame that is included in a payment request message *PReq* sent to the MCA 50 in step 14. In this way, no sensitive data will be readable on the local computer 44 nor transferred into the Internet.

The MCA 50 provides the necessary certificate and includes this in an authorization request *AuthReq* sent to the Payment Gateway 55 as step 15. The Payment Gateway 55 possesses the necessary key to decrypt the encrypted card-identification data provided in step 6, and provide a payment authorization request *AuthRes* to the MCA 50 in step 16, the MCA 50 then transmitting a payment request *PRes* to the LCA 44 in step 17 so the payment request can be performed on the LCA.

The card 31 can be removed at the end of the payment process. If a complementary request *ComplReq* (see below) is sent after the *PReq*, the reader 30 does not ask for card removal. Different messages are displayed on display 36 during further payment processing, even when an error occurs. An error interrupts the payment transaction and the reader 30 asks for card removal.

Steps 18-21: Further processing

After a payment transaction, the reader 30 can be asked to perform further processing by a *ComplReq* through this command. The Payment Gateway 55 can request for example an upgrade of the reader 30's software by initiating a download request and/or the display of a given message on the local computer screen.

Software Download

Fig. 5 illustrates a software download sequence. In this Figure, the numbers 1' to 8' are used to designate successive steps of the download sequence. Step 2' of Fig. 5 can be considered to be equivalent to step 22 of Fig. 4, in the case where the *ComplReq* is for software download. In this case, the *WakeUp* corresponds to the complete message contained in the *PRes* 17 response.

In response to a *Wakeup* in step 1', the LCA 44 exchanges a download request *DownloadReq* with the MCA 50 in steps 2' and 3', leading to downloading of software to the LCA 44. In steps 5' to 8', a *DWNReq* is exchanged between the LCA 44 and reader 30 leading to downloading of the software to the reader 30.

The software download application is protected by an RSA signature calculated using a private key generated by the controlling Organization who attests with this key that the software inside the reader 30 is qualified for performing the payment transactions according to the selected security protocol.

The memory 40 can allow for the application and system to be downloaded together, or separately. The files to be downloaded are grouped in a single file whose file name corresponds to the design of the reader 30.

As described previously, a buffer zone of the reader 30's E^2PROM serves as a buffer memory for temporarily storing system and/or application software transferred from the active E^2PROM during a download operation. In case of an interruption or failure of a download operation, a signal initiates reloading the software.

Installation

Because the reader 30 is controlled on the local computer 44 by a specific driver and a local computer application (LCA), the installation procedure is in two steps, driver installation and LCA installation, using for example a CD-ROM containing the software that is loaded into the computer.

Software Upgrade

Usually the software is automatically upgraded at the end of a standard transaction, like a payment. The system uses the fact that the reader 30 is connected to a management server for checking the actual software version that is installed in reader 30. If this version is lower than the current one on the server, it sends a download request to the reader 30, as described above. The user can also request independently a software download.

The reader 30 is designed to be attached to a PC through a USB connection 42/46. However, it not only offers smart card operations, but is also a powerful tool for cryptogram calculations. Because the card 31 and reader 30 are separate from the PC 44, the reader 30 provides a useful tool for securing any transaction that is undertaken on the PC. The reader 30's system and application software being downloadable, any application can be developed and downloaded into the reader 30. These applications can use the tool library that is available in the reader's downloaded system software.

The evolution of a reader 30 is linked to its software architecture which, as described before, is built according to a three shells model. The first shell, the boot shell 62, is fixed inside a given reader 30. After the manufacturing process, this boot shell 62 is never modified inside the reader 30. Over this boot shell 62 is the system shell 64 that contains the tool box including encryption/decryption functions. The functions of this systems library are compatible to all of the boot functions, regardless of the boot versions, inside a same model number. Most of the time an upgrade of this shell 64 will raise an upgrade of all of the applications. In other words, the reader 30 will be entirely upgraded (system and application shells 64, 66).

The applications are developed in such a way that they are independent from each other, but only dependent from the system shell 64 and/or possibly from the boot shell 62. This last dependence should be avoided, where possible. A new application always requires a system upgrade. But it should never require a significant boot upgrade, because this means that a new reader model would have to be created, compatible with this new application. The system shell upgrade is linked to the introduction of the entry points to the new application.

So when a new application is downloaded for the first time, the system must also be downloaded at the same time. Afterwards, the application can be downloaded separately. This will reduce the download duration for the upgrade of a single application. If an application becomes obsolete, it is possible:

- To upgrade this application. Usually this corresponds to a separate download of the application. Sometimes the system must be updated at the same time.
- To replace the application by another application. This does not mean that the system will be downloaded with a new one, because the entry points of the previous application can be reused.
- To erase the application. This means the entry points will be erased, so the system will be downloaded.

When a payment application from a given Organization is integrated into the current software, the reader 30 must be qualified by this Organization. Its software is signed by the Organization; the private key that controls the download operation is the property of the Organization. This procedure will be chosen for any application that requires a particular qualification for the reader integrity. Because there is a single public key that controls the software download, there is only one specific external Organization that can control the software integrity.

After the reader manufacturing process, any modification of the software must be downloaded. Because the software must be correctly signed for being accepted by the reader 30, the external Organization that has proprietary rights to the download private key is able to control and to qualify the integrity and the quality of the new software. Even if the software modification corresponds to the addition of a new application completely independent from their own applications, the external Organization is able to control that this new added application really does not alter their secure applications. The advantage of this feature is that the external Organization that controls the private key always keeps control of the software downloaded in the reader.

Digital Signature Process

With the combination of Smart Card, SET, Secure Readers & PC's, Governments and Corporate organisations want to expand the possibility to exchange secure data to everyone, for any transactions which are still processed with papers and analog signature, such as: VAT on line, E-Tax, E-voting, Letter of Credit, etc.

This means that sooner or later, everybody will have a Digital Identity Signature (DIS), which is the combination of a visit card, a pen and the traditional Identity Card. The delivery of the Digital Identity Signature (DIS) cannot be controlled by private companies. Today the Authorities allowed to issue the Digital Identity Signature (DIS) are controlled directly by the Governments involved, in order to guarantee the privacy of everyone.

Figs. 6 and 7 illustrate use of a device according to the invention which in this example is a keyboard unit connected as a terminal to a PC via a USB connection for one user to transmit a file with a digital signature (Fig. 6) and another user, the receiver, who uses a keyboard unit or other device according to the invention connected as a terminal to a PC via a USB connection, to authenticate the digital signature (Fig. 7), using a novel digital signature process.

The sending device and the receiving device are each associated with a separate and unique couple of keys, a private key and a public key. In the present example, the device/keyboard unit is associated with its own private key Pri_KC and public key Pub_KC; as well as a private key of the manufacturer Pri_KM and a public key of the manufacturer Pub_KM.

Generation of a digital signature by the sender is described first (Fig. 6), then authentication of the digital signature by the receiver (Fig. 7).

Generation of a Digital Signature by the Sender's Device/Keyboard Unit:

The sender wants to transmit a document called "File" 70 to the receiver, with a Digital Signature which cannot be repudiated. Any type of file format can be used for the document/data to be transmitted. The file 70 to be transmitted is compressed on the sender's PC by a hash coding into a 20 bytes (h-file)$_1$ file, which is transmitted to the sender's Keyboard Unit 34 via the USB link.

To identify the sender, there are 2 methods, either to use a Smart Card inserted in the keyboard unit's ICC interface 32 in which case the sender is identified by the PIN Code authentication of the Card, or the sender uses a PIN code of the Keyboard Unit 34 to identify the sender.

In both cases, the Card/Keyboard Unit Holder's Name "CHN" designated by B gives the identification information of the Sender on 26 Bytes. The Keyboard Unit 34 will then compress with a hash coding the sum of CHN and the (h-file)$_1$ into 20 Bytes. The result, (h-file)*$_1$, is then encrypted with the private key of the Keyboard Unit: Pri_KC to provide the function A which is Pri_KC(h-file)*$_1$.

To authentify the manufacturer of the Keyboard Unit 34 which is a trusted wallet device, the signature C (CS) of the Keyboard Unit is obtained by an 1023 bit RSA encryption of the private key of the manufacturer (Pri_KM) with the function D. The latter is obtained from the IFD, which stands for the Keyboard Unit's Identification Numbers (Serial number, software version, etc.) and the Public Key of the Keyboard Unit 34 (Pub_KC).

This produces a digital signature composed of the concatenated information R given by:

R = A + B + C + D, where
A = Pri_KC(h-file)$_1$
B = CHN (Card/Unit Holder Identification Code)
C = CS = Pri_KM(RSA 1023 bit)D
D = IFD + Pub_KC

The concatenated information or digital signature R is then transmitted to the sender's PC to be sent with the genuine (original) file 70 via Internet to the Receiver, the sent file S being represented by:

S = R+File(genuine).

Authentication of the Digital Signature by the Receiver's Device/Keyboard Unit:

As shown in Fig. 7, the Receiver receives via the Internet the sent file S. The PC's receiver extracts the genuine File 70 and compresses it with a hash coding into 20 Bytes: $(\text{h-file})_2$. This 20 bytes is then sent to the Receiver's Device/Keyboard Unit 34 with the concatenated information or digital signature R via the USB connection. The Receiver's Device/Keyboard Unit 34 completes these operations.

It decrypts with the Public Key of the Manufacturer (Pub_KM) and the Keyboard Unit Signature CS C with a 1023 bit RSA -1, and uses the received IFD from D to re-generate the Public key of the Keyboard Unit (Pub_KC).

It then decrypts with the Public Key of the Keyboard Unit (Pub_KC) and the received information A, Pri_KC$(\text{h-file})_1$*, with a 1023 bit RSA-1, and re-generates the file $(\text{h-file})_1$*.

In parallel, it generates the hashed file $(\text{h-file})_2$*, by hashing the concatenated file CHN B + $(\text{h-file})_2$ extracted from the received signature R.

Then it compares $(\text{h-file})_1$* and $(\text{h-file})_2$*:

If the compared files are identical, the Signature is considered to be successfully authenticated. The Receiver's Keyboard Unit 34 displays on its secure display 36 "Signature OK", also displaying the Card Holder Information "CHN". Simultaneously, this information is transmitted to the Receiver's PC 44, which also displays on the unsecured PC's screen 47 the same information.

If the files are different, the Signature is considered to be wrong. The Keyboard Unit 34 displays on its secure display 36 "Wrong Signature". Simultaneously, this information is transmitted to the Receiver's PC 44, which will also display on the unsecured PC's screen 47 the same information.

This digital signature process is based on the concept that no public key needs to be sent between the sender and the receiver. On both sides, we use the private key and the public key of the trusted wallet device or keyboard unit, encrypted into its silicon, as well as the private key and the public key of the keyboard unit's manufacturer. The privacy and the secrecy of the sender can be protected, because no potential identification during the transmission can be performed. During transmission, the public key Pub-KC is encrypted as part of function D, and is not accessible as such.

A Smart Card can be used if it is necessary to recognise the Card owner via its PIN code. The digital signature can use the secure Device/Keyboard Unit to process the PIN code to identify the Keyboard Unit's owner.

Any type of file format can be used for the document/data 70 to be transmitted. Any PC platform can be used (Windows 98, Millenium, 2000 & iMac).

In comparison with other methods to sign a file like PGP, this digital signature is capable of:

- Keeping the privacy of any sender on Internet: No public keys are sent, and the sender cannot be identified on Internet when any hacker attempts to intercept the message.
- Signing any messages with a non repudiation feature, when using the Device with a Smart Card.
- Authenticating the sender's Identity, with the use of the PIN Code, combined with the Device's couple of RSA's keys.
- Guaranteeing a unique signature for any user, like an Identity Card: Each Device/Keyboard Unit has its own unique couple of keys, each being different between 2 Devices/Keyboard Units (also compatible with discrete reading devices).
- In addition to the given example of Secure Keyboards, this digital signature can be applied to and is compatible with other reader devices: discrete reader devices, GSM Mobiles, TV-remote controls, end-to-end routers and servers, Digital TVs, Gambling, Virtual Site for Governments, etc.
- Protecting PCs against reception of corrupted files (with virus, like Trojan Horses).
- Making sure that no duplication of signatures is possible, because the logical information is tied to unique silicon firmware (i.e. the Device's firmware).
- Making sure that the signature cannot be manipulated throught the PC's Bios (basic input-output system).

The fact that each device is associated with its own unique couple of encryption keys (a private key and a public key) means that all communications and operations with each device are unique and can be distinguished from those made with other devices even when using the same ICC card. In particular, operations using the same ICC card with different devices held by the same user can be discriminated.

Generalities

It can be seen from the above that the invention has provided a versatile device ensuring entire security for the user, and which can be integrated in existing or future protocols directed to secure transactions in communications networks, in particular under control of an Organization that possesses the reader's private key. Furthermore, the described reader can be used for many other secure transactions with or without the insertion of a card, including e-banking, e-tax and secure e-mail. The device can be embodied in many forms. A particular embodiment in the form of a secure keyboard is further described in copending patent application PCT/IB01/00896. Various modifications may be made to the described examples of hardware and software without departing from the scope of the appended claims.

CLAIMS

1. A device connectable as a terminal of a communications network for carrying out secure transactions or any PIN code processes in the network, in particular in connection with e-commerce, e-banking, e-tax, e-mail, telecommunications and restricted access systems, the device comprising:

a keypad, a display for displaying messages related to a secure transaction in progress, a microprocessor, and a memory arranged to store software in three shells, a boot shell, a system shell and an application shell, wherein the boot shell contains non-downloadable ground software that manages software download into the system shell and the application shell;

the device being associated with software downloadable into or downloaded in the memory, namely:

- system shell software containing software that manages the application shell, an ASN library and an encryption/decryption toolbox, and

- application shell software containing software that manages applications for secure transactions or any PIN code processes, comprising an application that is arranged, when the software is downloaded into the memory and the device is connected to the communications network, to encrypt within the device a user identification code keyed in via the keypad and output the encrypted code, without the keyed-in user identification code being accessible from outside the device.

2. The device of claim 1, wherein said application is arranged to perform the following operations internally of the device:

- momentaneously store, in a network-inaccessible part of the memory, a user identification code keyed in via the keypad;
- encrypt the momentaneously-stored keyed-in user identification code and erase it from the memory soon after encryption; and
- output the encrypted user identification code from within the device to the communications network.

3. The device of claim 1 or 2, comprising a card reader for an ICC card, the device being arranged to enable card identification and acceptance on an ICC inserted in the reader followed, if the card is accepted, by said operations to establish transaction acceptance.

4. The device of claim 3, wherein at least a part of the card identification and acceptance operations and/or of said transaction acceptance operations is performed using a processor incorporated in an ICC inserted in the device.

5. The device according to any preceding claim, wherein the memory comprises a buffer zone which serves as a buffer memory for temporarily storing previously-loaded system and/or application software during a download operation, the memory being associated with means, responsive to a signal indicating interruption or failure of a download operation, for reloading system and/or application software.

6. The device according to any preceding claim, wherein the memory comprises a non-volatile memory, in particular an E^2PROM, for storing the downloadable software, and a RAM which includes said network-inaccessible part of the memory.

7. A device according to any preceding claim incorporated in: (a) a hand-holdable external card reader connectable to a personal computer; (b) a discrete keyboard of a personal computer, connectable to a personal computer; (c) a portable personal computer; (d) a Set Top Box of a digital television receiver; (e) a fixed telephone set; (f)

a mobile communications device such as a mobile telephone; or (g) a point-of-sale vending device.

8. A device according to claim 7 in the form of a discrete keyboard of a personal computer having an ICC interface and associated with a keyboard matrix selectively connectable between a normal operation mode for communicating keycodes to the Bios of a locally connected computer, and a secure transaction mode for carrying out secure transactions or any PIN code processes with or without an ICC inserted in the ICC interface.

9. A cardholder system for carrying out secure transactions in a communications network, comprising a device according to any preceding claim, associated with a local computer application (LCA), such as a PC, to which the device is connected, preferably by an USB connector.

10. The cardholder system of claim 9, wherein the LCA stores software including a Dynamic Link Library (DLL) for for managing all transmissions between the local computer and a connected device.

11. The cardholder system of claim 9 or 10 which is configured to produce an encrypted digital signature derived from a private key (Pri_KC) associated with the device (A), a user identification code (B), a private key (Pri_KM) associated with the keyboard unit's manufacturer (C), and a public key (Pub_KC) of the keyboard unit.

12. The cardholder system of claim 11, which is arranged to encrypt the public key (Pub_KC) of the keyboard unit before transmission to produce a transmittable digtal signature containing no public key.

13. The cardholder system of claim 12, which is configured to authenticate said digital signature by decrypting with a public key of the manufacturer (Pub_KM) the public key (Pub_KC) of a sending keyboard unit or device.

14. A communications network in which encrypted data is communicated between a card issuer system, merchant site applications, a payment gateway and a device according to any one of claims 1-9 or a cardholder system according to any one of claims 9 to 13, wherein the device is connected to the communications network to form a terminal for carrying out secure transactions or PIN code processes in the network.

15. The communications network of claim 14 wherein the card issuer system, merchant site applications, and payment gateway exchange electronic certificates according to a protocol such as SET.

16. A computer program on a computer readable medium for carrying out secure transactions in a communications network to which a device as claimed in any one of claims 1-8 is connected via a local computer application, comprising said system shell software and application shell software downloadable into the device's memory, and software downloadable into the local computer application for managing all transmissions between the local computer and a connected device.

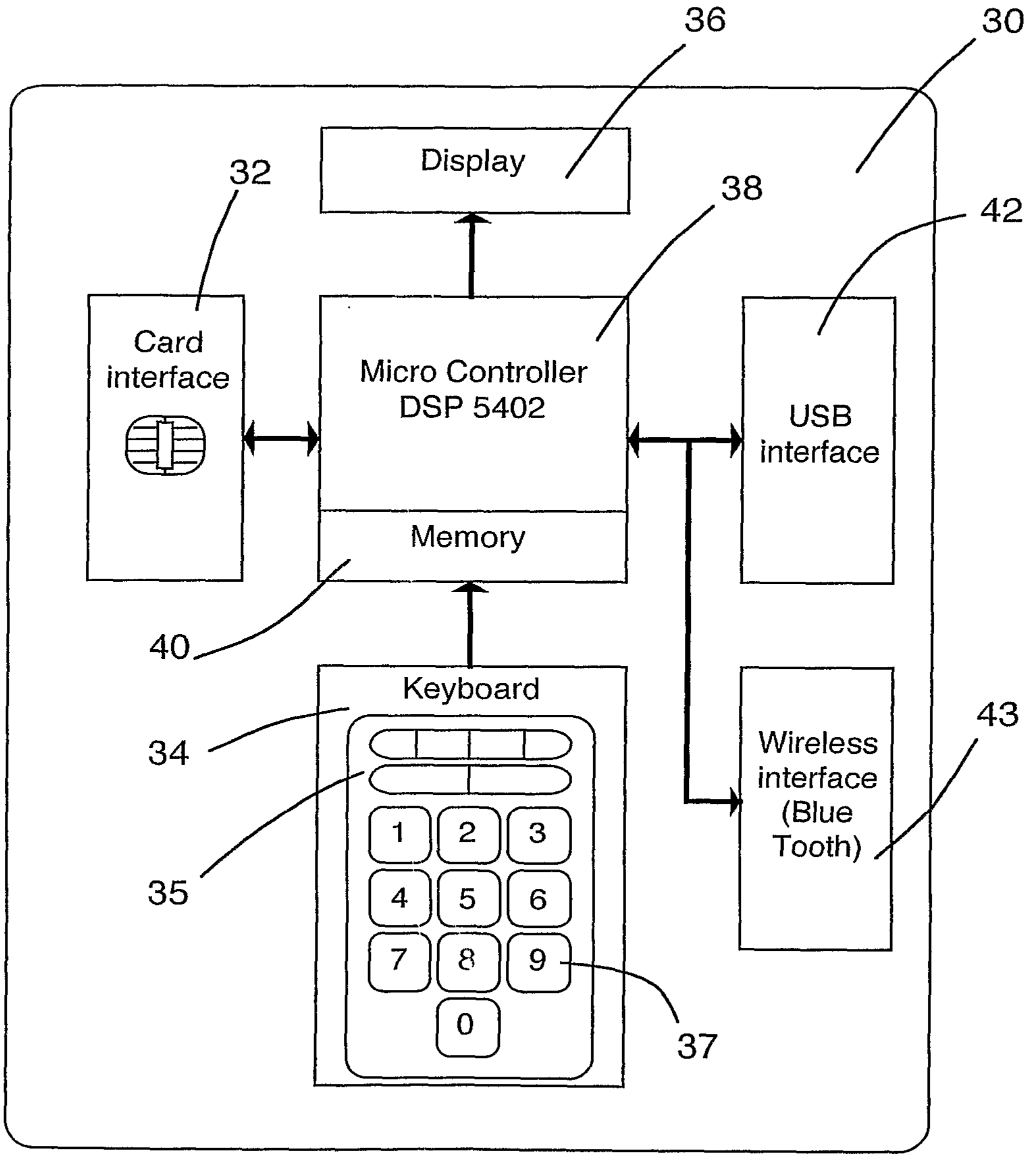

36
30
Display
32
38
42
Card interface
Micro Controller DSP 5402
USB interface
Memory
40
Keyboard
34
43
Wireless interface (Blue Tooth)
35
1
2
3
4
5
6
7
8
9
0
37

Fig. 1

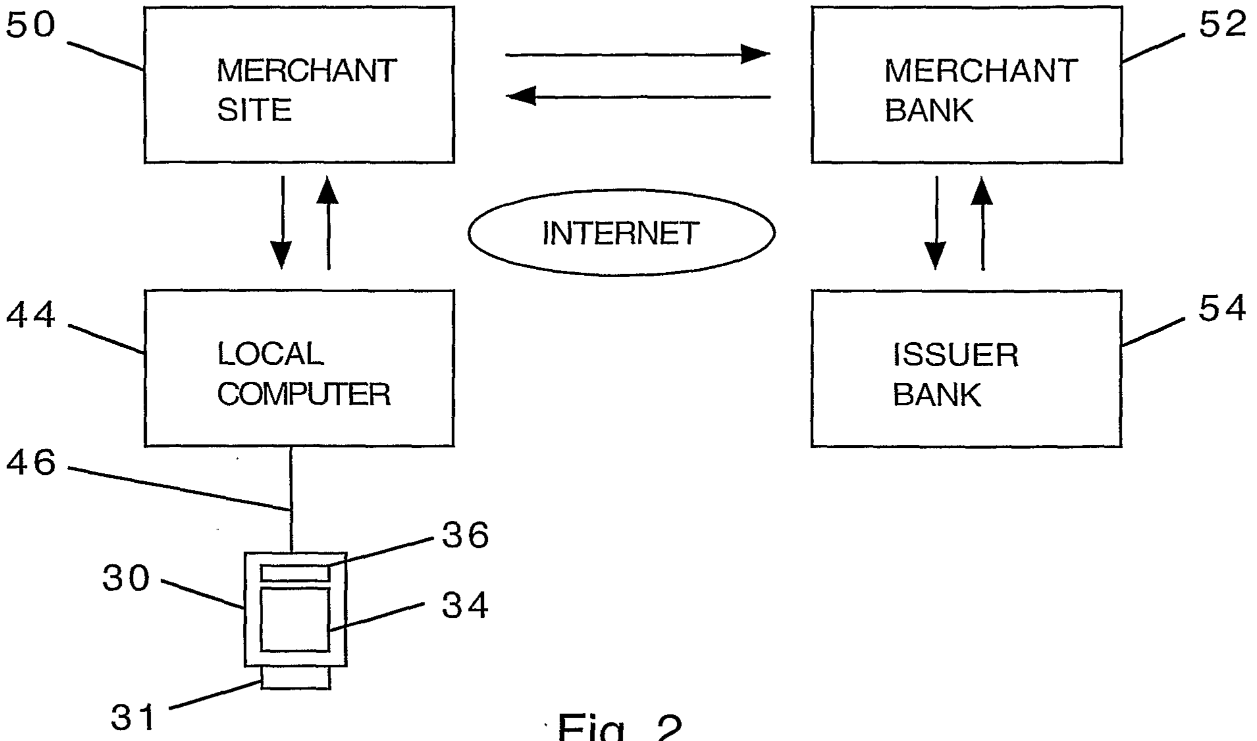
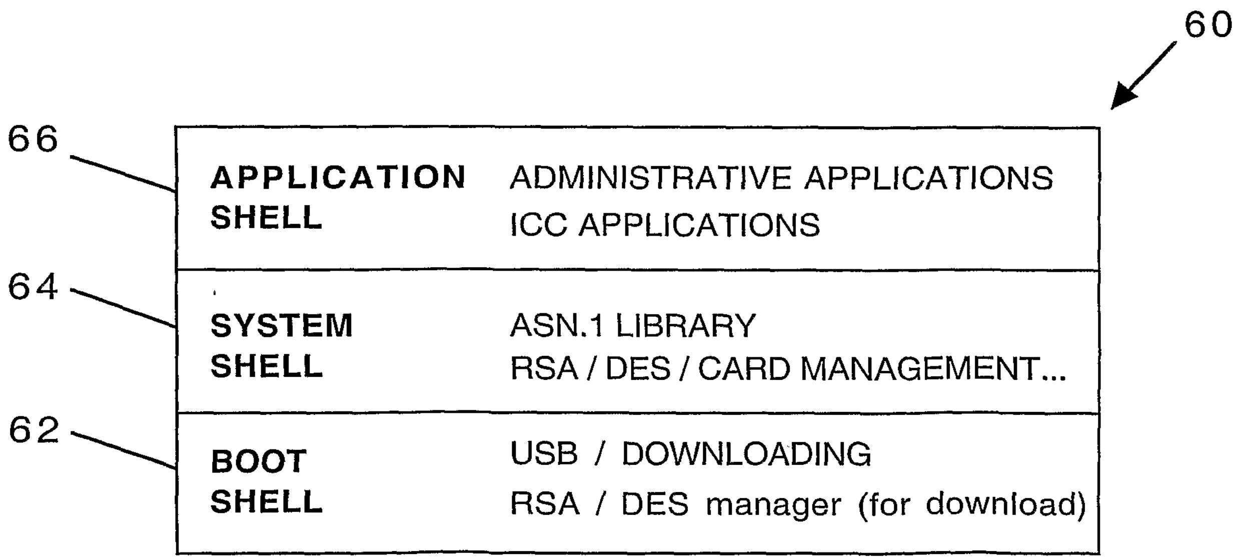


Fig. 2



APPLICATION SHELL	ADMINISTRATIVE APPLICATIONS ICC APPLICATIONS
SYSTEM SHELL	ASN.1 LIBRARY RSA / DES / CARD MANAGEMENT...
BOOT SHELL	USB / DOWNLOADING RSA / DES manager (for download)

Fig. 3

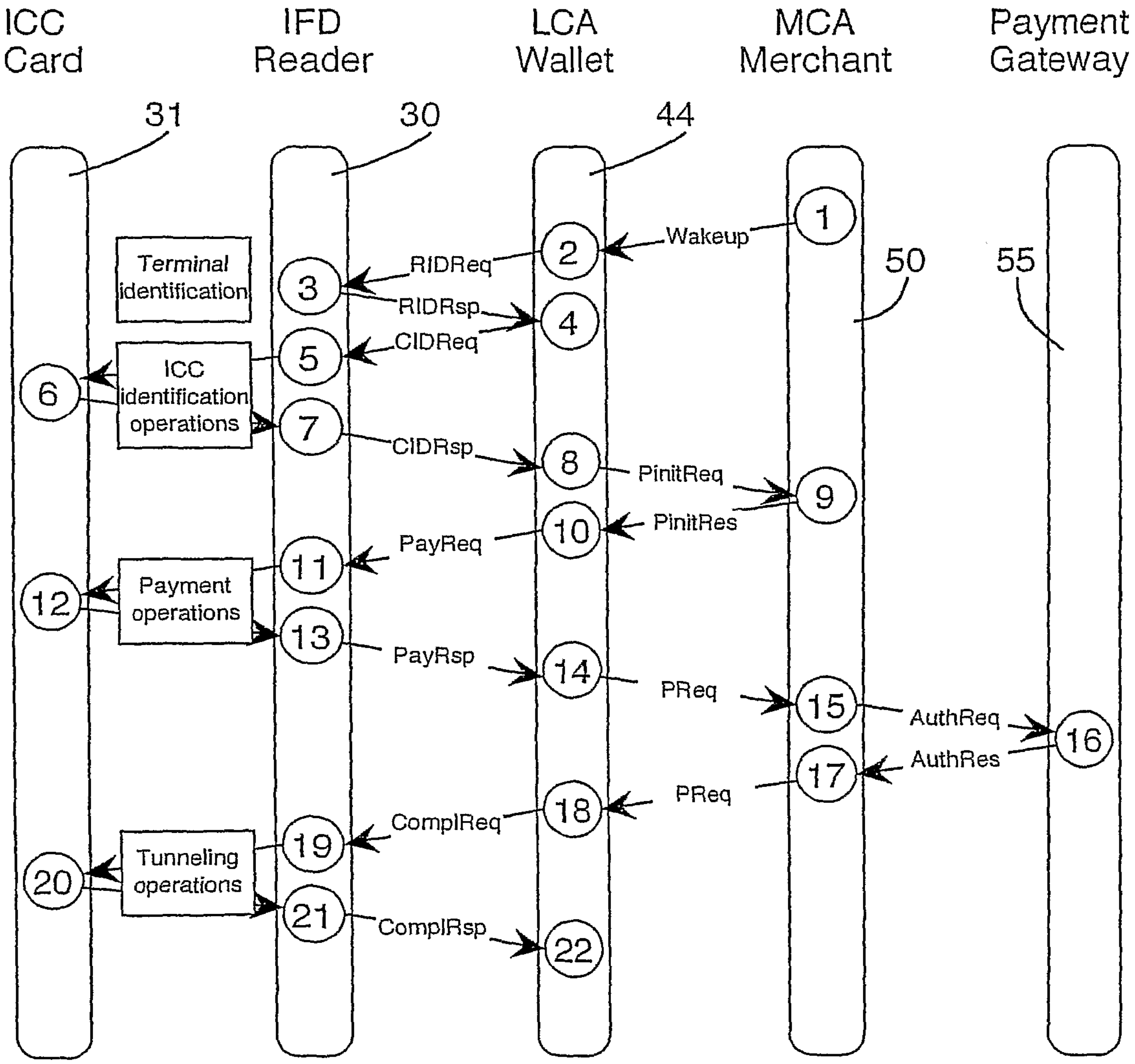

ICC
Card
IFD
Reader
LCA
Wallet
MCA
Merchant
Payment
Gateway
31
30
44
50
55
1
Wakeup
2
RIDReq
3
Terminal identification
RIDRsp
4
CIDReq
5
ICC identification operations
6
7
CIDRsp
8
PinitReq
9
PinitRes
10
PayReq
11
Payment operations
12
13
PayRsp
14
PReq
15
AuthReq
16
AuthRes
17
PReq
18
ComplReq
19
Tunneling operations
20
21
ComplRsp
22

Fig. 4

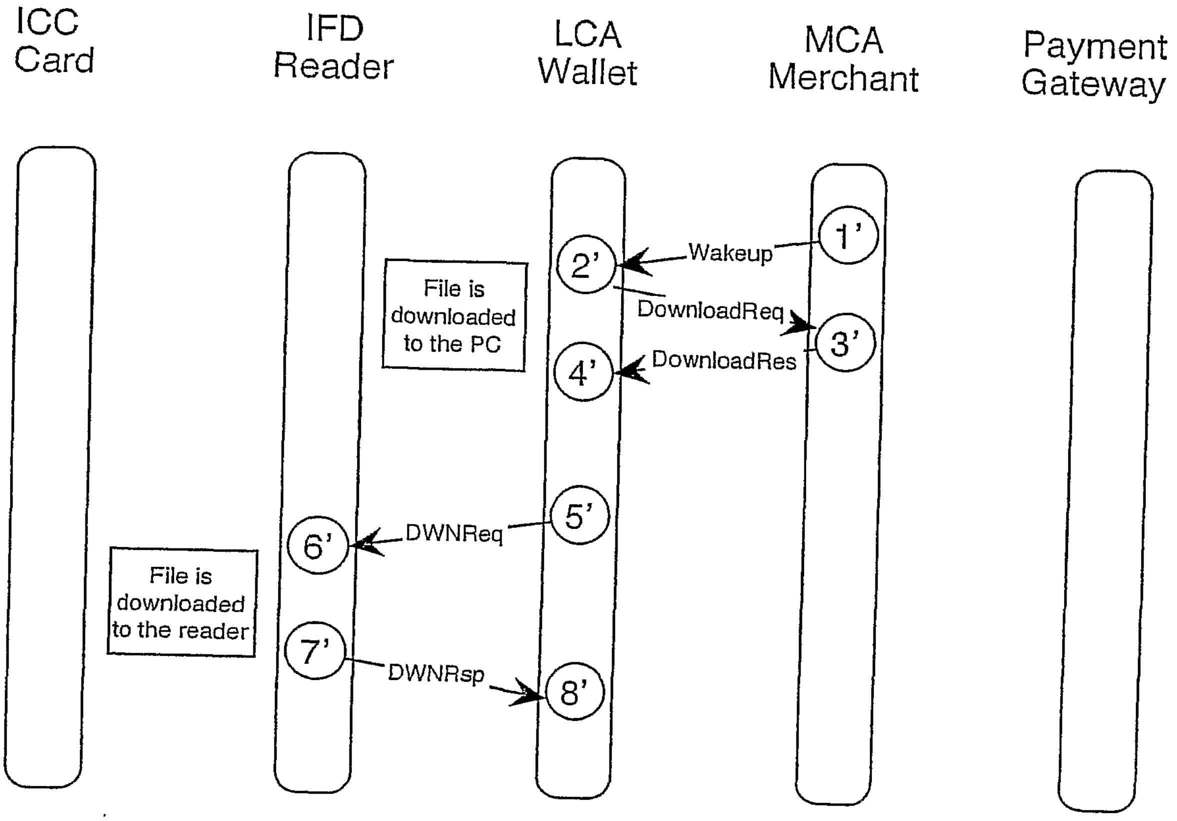

ICC
Card
IFD
Reader
LCA
Wallet
MCA
Merchant
Payment
Gateway
1'
Wakeup
2'
File is downloaded to the PC
DownloadReq
3'
4'
DownloadRes
5'
DWNReq
6'
File is downloaded to the reader
7'
DWNRsp
8'

Fig. 5

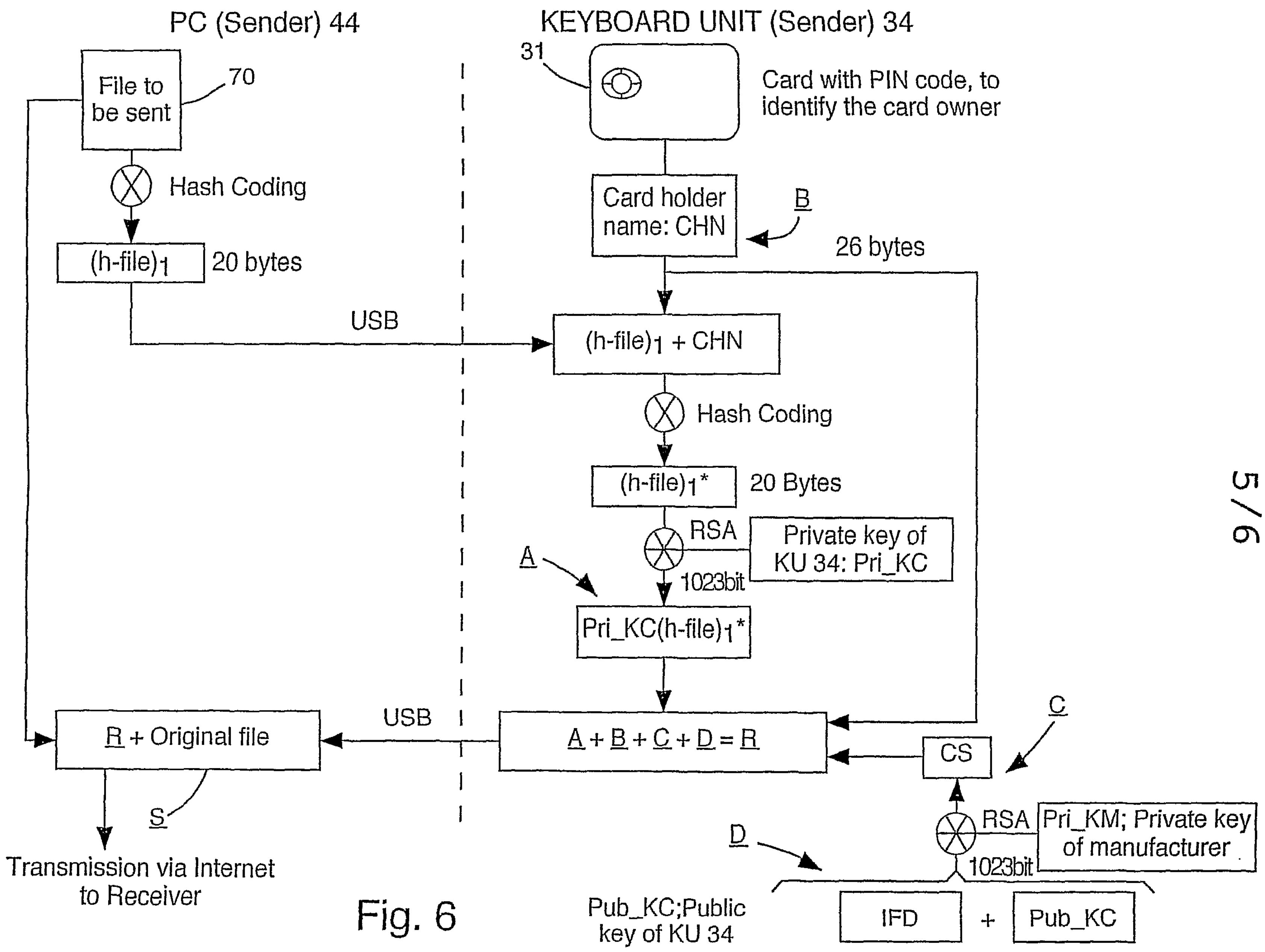

PC (Sender) 44
KEYBOARD UNIT (Sender) 34
31
Card with PIN code, to identify the card owner
File to be sent
70
Hash Coding
(h-file)$_1$
20 bytes
USB
Card holder name: CHN
B
26 bytes
(h-file)$_1$ + CHN
Hash Coding
(h-file)$_1$*
20 Bytes
RSA
Private key of KU 34: Pri_KC
1023bit
A
Pri_KC(h-file)$_1$*
A + B + C + D = R
USB
R + Original file
S
Transmission via Internet to Receiver
CS
C
RSA
Pri_KM; Private key of manufacturer
1023bit
D
Pub_KC;Public key of KU 34
IFD
+
Pub_KC

Fig. 6

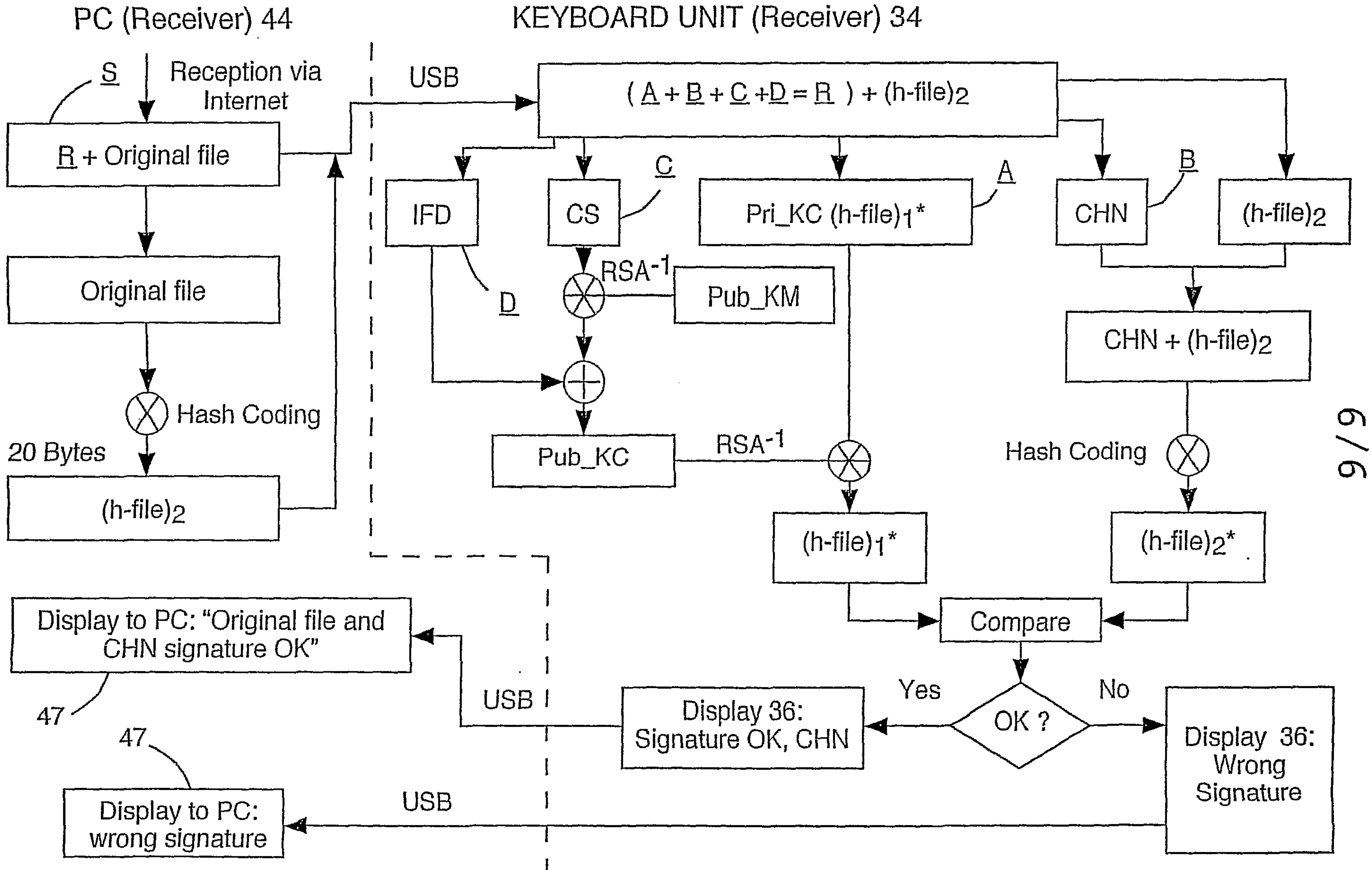

PC (Receiver) 44
KEYBOARD UNIT (Receiver) 34
S
Reception via Internet
R + Original file
Original file
Hash Coding
20 Bytes
(h-file)2
USB
(A + B + C + D = R) + (h-file)2
IFD
D
CS
C
RSA-1
Pub_KM
Pri_KC (h-file)1*
A
CHN
B
(h-file)2
CHN + (h-file)2
Hash Coding
Pub_KC
RSA-1
(h-file)1*
(h-file)2*
Compare
OK ?
Yes
No
Display 36: Signature OK, CHN
Display 36: Wrong Signature
USB
Display to PC: "Original file and CHN signature OK"
47
47
USB
Display to PC: wrong signature

Fig. 7

INTERNATIONAL SEARCH REPORT

In ıal Application No
P('IB 01/01120

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G07F19/00 H04L9/32 G07F7/10

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04L G07F G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WO 98 07255 A (INFORMATION RESOURCE ENGINEERI) 19 February 1998 (1998-02-19) cited in the application page 4, paragraph 4 - paragraph 5 page 8, line 16 -page 9, line 19 page 16 -page 18, line 5 page 19, paragraph 2 -page 20, paragraph 2 page 22, paragraph 2 ---	1-15
Y	WO 98 49658 A (CONKLIN FREDRICK SIDNEY ;PRINGLE STEVEN JOHN (US); BERG MICHAEL J) 5 November 1998 (1998-11-05) page 8, paragraph 2 page 15, paragraph 2 - paragraph 3 page 18, paragraph 3 -page 19, paragraph 1 page 20, paragraph 3 - paragraph 4 page 29, paragraph 3 --- -/--	1-7,9-15

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search 2 October 2001	Date of mailing of the international search report 19/10/2001
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Authorized officer Wolles, B

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
Y	EP 0 587 375 A (ALGORITHMIC RES LTD) 16 March 1994 (1994-03-16) cited in the application	8
A	column 1, line 1 -column 3, line 21 column 4, line 32 -column 5, line 33 column 8, line 5 - line 13 ---	1-7,9-15
A	US 4 731 842 A (SMITH PETER R) 15 March 1988 (1988-03-15) column 2, line 49 -column 3, line 31 column 8, line 23 - line 30 claims 1-7 ---	
A	EP 0 763 791 A (HEWLETT PACKARD CO) 19 March 1997 (1997-03-19) cited in the application column 2, line 50 -column 4, line 9 ---	
A	WO 98 59327 A (DIGITAL EQUIPMENT BCFI AB ;HEDIN BENGT (SE); JANSSON KJELL (SE); M) 30 December 1998 (1998-12-30) page 4, line 24 -page 5, line 17 ---	
A	GB 2 274 523 A (PATNI CHANDRA KAMAR) 27 July 1994 (1994-07-27) cited in the application page 1 figure 1 ---	
A	US 5 336 870 A (HUGHES THOMAS S ET AL) 9 August 1994 (1994-08-09) cited in the application abstract ---	
A	DE 43 25 459 A (C2S GMBH CRYPTOGRAFISCHE SICHE) 9 February 1995 (1995-02-09) claims 1-16 -----	

INTERNATIONAL SEARCH REPORT

›rmation on patent family members

In ›nal Application No
PCT/IB 01/01120

Patent document cited in search report		Publication date	Patent family member(s)			Publication date
WO 9807255	A	19-02-1998	US	5778071	A	07-07-1998
			AU	726397	B2	09-11-2000
			AU	4147097	A	06-03-1998
			EP	0916210	A1	19-05-1999
			WO	9807255	A1	19-02-1998
WO 9849658	A	05-11-1998	US	6282522	B1	28-08-2001
			AU	7272698	A	24-11-1998
			EP	1003139	A2	24-05-2000
			EP	1023705	A1	02-08-2000
			WO	9849658	A1	05-11-1998
			US	6105008	A	15-08-2000
EP 0587375	A	16-03-1994	IL	103062	A	04-08-1996
			EP	0587375	A2	16-03-1994
			GB	2267986	A ,B	22-12-1993
			SG	43927	A1	14-11-1997
			US	5406624	A	11-04-1995
US 4731842	A	15-03-1988	GB	2168514	A	18-06-1986
			DE	3585439	A1	02-04-1992
			EP	0186981	A2	09-07-1986
			JP	61139878	A	27-06-1986
EP 0763791	A	19-03-1997	EP	0763791	A1	19-03-1997
			US	5920730	A	06-07-1999
WO 9859327	A	30-12-1998	SE	511507	C2	11-10-1999
			AU	8044798	A	04-01-1999
			JP	2000507380	T	13-06-2000
			SE	9702216	A	11-12-1998
			WO	9859327	A1	30-12-1998
GB 2274523	A	27-07-1994	NONE			
US 5336870	A	09-08-1994	US	5754655	A	19-05-1998
DE 4325459	A	09-02-1995	DE	4325459	A1	09-02-1995